

Connect/C++ Connect/Java

株式会社 数理システム
応用技術部
関川拓也

概要

S-PLUS は S-PLUS 以外の外部アプリケーションとの間の通信のためにさまざまな機能を備えている . その中で新たに S-PLUS 6.1 for Windows に追加されたインターフェースである Connect/C++ と Connect/Java を簡単に紹介する .

S-PLUS for Windows 外部インターフェースの変遷

S-PLUS for Windows の外部プログラム向けインターフェースとしては、これまで以下のようなものが提供されてきた .

.C()/Fortran()
DDE
Automation
S+API

S-PLUS 6.1 for Windows にはこれらに加えてさらに新たなインターフェースが 2 つ追加されている .

Connect/C++
Connect/Java

これらの外部公開インターフェースの特徴を簡単に説明する .

.C()/Fortran()

それぞれ、C/C++言語、Fortran 言語で記述された関数をコールするための S-PLUS の組み込み関数である。初期の頃の S-PLUS からある関数だが、最も手軽な機能拡張手段であり、現在でも使われることは多いと思われる。Windows 版は DLL のロード関数(dll.load())と共に使うのが一般的である。

DDE

Windows95 以前の Windows のアプリケーション間通信で一般的な手法だった DDE(Dynamic Data Exchange)を使ったものである。現在は Automation があるため、使われることは少ないだろう。

Automation

Automation は DDE よりも新しく、使い勝手の良い Windows 上でのアプリケーション間通信の手段である。Automation は言語非依存という大きな特徴があるため、VisualBasic はもちろん、Delphi, Perl, Python, Lisp などの言語からも利用可能である。

S+API

S-PLUS 6.0 以前の C/C++言語向けのインターフェースである。前述の.C()とは逆に、外部プログラムから S-PLUS 内の関数をコールするための機構である。S-PLUS Engine DLL のみの利用方法であるため高速な反面、手続きが煩雑、グラフが描けないなどの欠点もある。

Connect/C++

S-PLUS 6.0 for Windows から装備された C++向けのクラスライブラリである。S+API に代わるものと考えればいだろう。利用の手軽さは S+API よりもはるかに簡単になった。ただし S+API と同じくグラフ描画機能等はない。また、S+API とのソースコードレベルでの互換性は全くない。

Connect/Java

Unix 版にあった Java 連携機能と同じものである。Windows 版では S-PLUS 6.1 からようやく利用可能になった。S-PLUS から Java の関数をコールすることも、その逆もできる。また Connect/C++ と異なりグラフ描画機能も備えている。

Connect/C++と Connect/Java の詳細

Connect/C++

Connect/C++は S-PLUS6.0 以前にあった S+API がクラスライブラリ化されたもの、と考
えてよいだろう。クラスライブラリ化されたことで使い勝手はかなり向上した。

たとえば、S-PLUS 関数の

```
x <- rnorm(10)
```

にあたる処理をこれまでの S+API で行うと、このような単純な処理にも関わらず 50 行近
くもコードを書く必要があった。

しかし同様の処理を Connect/C++を使って書いた場合は以下のように非常に単純になる。

```
#include <iostream.h>
#include "sconnect.h"

void main(int argc, char *argv[])
{
    // 初期化
    CSPengineConnect sp;
    sp.Create(argc, argv);

    // S-PLUS Engine コール
    sp.SyncParseEval("x <- rnorm(10)");

    // 結果表示
    CSPnumeric x = sp.get("x");
    for (int i=0; i<x.length(); i++) cout << (double)x[i] << " ";
}
```

```
0.757121 0.176686 -0.632562 0.0911522 1.01901 1.47113 3.10838 0.17567 1.06679 0.
450745
```

このプログラムで使用している `SyncParseEval()` メソッドは、任意の S 言語の文字列表現を受け取って実行するものだ。例えばコンソールから S 言語で記述された文字列を受け取り、`SyncParseEval()` で実行するようなプログラムを書けば、コマンドライン上での S 言語実行環境が出来上がる。

Connect/Java

その名前から連想されるとおり、Connect/C++ の Java 版であると思ってよいだろう。ただし Connect/C++ と異なり、グラフ描画機能も持っている。

先ほどのサンプルと同じく、`rnorm(10)` を行うプログラムを Connect/Java を使って書くと以下のようなになる。

```
import com.insightful.splus.*;

public class RNormExample {
    public static void main(String [] args) {
        try {
            String expression = "rnorm(10)";
            SplusDataResult result =
                SplusUserApp.eval(expression + "\n",
                true, false, false, false, false);
            System.out.println(result.getOutput());
        }
        catch (Exception e) {
            System.out.println(e);
        }
        System.exit(0);
    }
}
```

```
[1] -0.20281717 -0.35912965 -0.38225727  1.59789958 -0.02346776
[6] -0.39354119 0.33727084 -1.06062044  0.58332922  0.61470433
```

グラフ利用のサンプルは少々長いため、デモでの実行結果のみ示す。

まとめ

S-PLUS for Windows に新たに加わった外部アプリケーション向けインターフェースである Connect/C++と Connect/Java についての概略を説明した．これらのインターフェースを活用することでさらに S-PLUS の応用範囲が広がるだろう．